SIGNAL PROCESSING IMPROVEMENTS TO LOCALIZATION FOR AUTONOMOUS VEHICLES

A Thesis

by

SAMUEL TODD FLANAGAN

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Jean-Francois Chamberland
Committee Members,	Krishna Narayanan
	Swaminathan Gopalswamy
Head of Department,	Miroslav Begovic

May 2021

Major Subject: Electrical and Computer Engineering

ABSTRACT

Precise localization is a fundamental part of vehicular autonomy. Current implementations rely on expensive sensor arrays and favorable conditions to accurately localize a vehicle. As autonomous vehicles progress toward production models, these expensive sensors will be replaced by low-cost alternatives. Additionally, production autonomous vehicles will need to operate reliably in a range of conditions. Localization algorithms will need to accurately perform despite the increased noise due to these factors. This thesis examines mathematical improvements to existing localization techniques in an effort to increase their reliability in adverse conditions. The effectiveness of these improvements is evaluated using numerical simulations as well as testing with real-world data.

DEDICATION

To my mother, my father, and my grandparents for supporting me throughout my education.

ACKNOWLEDGMENTS

I would like to thank Dr. Jean-Francois Chamberland and Siddharth Agarwal for guiding me through my graduate studies. I would also like to thank Will Flanagan for proofreading this thesis.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a thesis committee consisting of Professor Jean-Francois Chamberland and Professor Krishna Narayanan of the Department of Electrical and Computer Engineering and Professor Swaminathan Gopalswamy of the Department of Mechanical Engineering. The simulations in Section 3.2.3 were conducted with the advanced computing resources provided by Texas A&M High Performance Research Computing. The data analyzed in Section 3.3.2 and localization testing performed in Section 3.3.3 were provided by Siddharth Agarwal and Ford Autonomous Vehicles LLC. All other work conducted for this thesis was completed by the student independently.

Funding Sources

There is no financial support to report.

NOMENCLATURE

AV	Autonomous Vehicle
MI	Mutual Information
NMI	Normalized Mutual Information
SLAM	Simultaneous Localization and Mapping
GPS	Global Positioning System
SNR	Signal-to-Noise Ratio
ML	Maximum Likelihood
EKF	extended Kalman filter
PMF	Probability Mass Function
CDF	Cumulative Distribution Function
IMU	Inertial Measurement Unit
RMSE	Root Mean Squared Error
LiDAR	Light Detection and Ranging

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION AND LITERATURE REVIEW

Autonomous vehicle research receives generous amounts of attention and funding. The promise of cars that drive themselves and the convenience they would provide have led to the field's high valuation. Convenience, however, is not the only benefit of autonomous vehicles. According to the National Highway Traffic Safety Administration, over 35,000 people died in the US in 2018 alone due to motor vehicle crashes [1]. Though the technology has not yet matured, autonomous vehicles could significantly reduce such deaths because computer agents are not susceptible to many leading causes of traffic fatalities such as alcohol, distracted driving, and speeding [2].

Production autonomous vehicles, widely available to consumers, are critical in realizing these benefits. Current research ventures rely on expensive prototypes with sophisticated sensor arrays that provide data with high accuracy, but are prohibitively costly for production vehicles. Cost-effective solutions will need to be developed in order to bring autonomous vehicles to a broad consumer market. Relatively inexpensive sensors are necessary for wide-scale adoption and, therefore, engineering solutions must be robust to noisy data. This thesis focuses on improving existing localization algorithms to create more reliable solutions for future production autonomous vehicles equipped with affordable sensing units.

1.1 Localization

Localization is the process by which an autonomous vehicle determines its location within its surroundings. Reliable localization with centimeter-scale accuracy is critical to autonomous vehicles. Systems based on satellites and cellular infrastructures alone are inadequate for this application because they provide coarse location information and are unreliable in urban settings with limited sky clearance. Recognizing this need, researchers and engineers have developed enhanced methods of localization, often turning to LIDAR technologies. In this broad context, there are two main localization strategies: simultaneous localization and mapping (SLAM) and localization with a prior map. Since autonomous vehicles in early deployment are confined to known geographical areas, we focus on the latter scenario. Nevertheless, we review both approaches briefly to better contextualize our contribution.

1.1.1 Simultaneous Localization and Mapping

Simultaneous localization and mapping (SLAM) dominates much of the existing research in the autonomous vehicle (AV) space. Over the past two decades many implementations have been explored. For instance, Dissanayake et al. used an estimation-theoretic or Kalman filter based strategy to solve the SLAM problem [3]. We note that multiple competing implementations such as DP-SLAM [4], Atlas [5], and Graph SLAM [6], are available in the literature. One of the most successful SLAM algorithms, a variant of Graph SLAM, achieves "reliable real-time localization with accuracy in the 10-cm range" [7]. This algorithm was subsequently improved by using probabilistic maps as a representation of the surroundings that "increased robustness to environmental changes and dynamic obstacles," while maintaining comparable accuracy [8]. We implicitly utilize some aspects of the SLAM framework in our localization strategy, especially while performing analysis of overall systems; yet these aspects are ancillary to our research and its focus on reliable data acquisition. Thus, SLAM lies outside the scope of the thesis.

It is worth mentioning that previous publications explore how to perform SLAM using inexpensive cameras (visual SLAM). However, existing solutions struggle in challenging conditions [9]. It is likely that improving image acquisition would help the reliability of localization techniques under such conditions within SLAM; but our work focuses on a different type of localization strategy, as described below.

1.1.2 Localization with a Prior Map

In this thesis, we are primarily concerned with localization strategies that make use of prior maps. Such a map is built over multiple passes and thus is assumed to be an accurate (essentially noiseless) top-down view of the environment. Given such a prior map, the localization process reduces to matching the current sensor input from the vehicle with a section of the map and inferring the vehicle's position. In practice, techniques such as extended Kalman filtering (EKF) are used to reduce the search space of possible map sections. An EKF is used in localization to estimate vehicle position based on previous state information [10]. This estimate is then used as a starting location in the prior map to begin searching for the map section that matches the current sensor input. In this context, we center our efforts on improvements in the image matching task for localization.

1.2 Mutual Information

An information measure that plays an important role in our impending discussion is mutual information (MI). Loosely speaking, MI quantifies the amount of shared information between two random variables. This criterion can also be extended to empirical measurements, using the empirical joint distribution of two groups of data. In practice, MI can be employed to determine whether two images match, and it features some robustness to illumination and noise. Much of the current research on MI applications focuses on its use for medical image registration [11]. In this field, researchers seek to match images or features within an image with a computer model or physical locations [12]. The use of MI for image registration was spearheaded by Collingnon, Maes, Viola, and Wells [13, 14, 15, 16]. Their work made the technique popular, as it performed rigid registration of multi-modality images with notable success. Rigid registration assumes images only need to be rotated and translated to find their correct matches, while non-rigid registration requires "some localized stretching of the images" [17]. Our work more closely relates to rigid registration because we assume the transformation needed to match the vehicle's current sensor input with a section of map is calculated during calibration. Multi-modality images are images generated by different means such as "computed tomography (CT), magnetic resonance (MR), and photon emission tomography (PET)" [14]. These images are inherently different due to the properties of each modality and thus present a challenge for image registration. Robustness to image modality is an attractive characteristic of MI in regards to localization because the vehicle's sensor input will likely vary greatly from the prior map due to differing conditions. Normalized mutual information (NMI), a normalized version of MI, was later proposed by Studholme et al. [18]. This measure shows increased robustness to outliers and differences in image overlap. We use this formulation of NMI in our work.

1.2.1 NMI and Localization

The characteristics of NMI that make it an effective technique for medical image registration also make it suitable for localization. It has shown robustness to obstructions, lighting [19], outliers, and image overlap [18]. Because of this, NMI has entered into localization research. Walcott and Eustice used NMI and a monocular camera to perform localization with a 3D map [20]. In their implementation, they transformed sections of the map into the perspective of the camera and matched them with camera images using NMI. We examine the reverse approach in our work by transforming camera images into the perspective of the map and matching them. Another implementation of NMI-based localization was proposed by Castorena and Agarwal [21]. They used a map of LiDAR reflectivity edges that removed the need for post-factory laser reflectivity calibrations and slightly outperformed the state of the art in localization performance.

The NMI measure does not account for noise levels, though it is somewhat robust to noise [19]. In fact, using NMI as an image matching technique for localization makes the assumption (intrinsically) that all information in both the map and current sensor input is of equal quality. We will examine the validity and impact of this assumption in later sections.

2. IMAGE ACQUISITION IN AUTONOMOUS VEHICLES

2.1 Pinhole Camera Model

We begin our exposition with the description of a simplified image acquisition model for autonomous vehicles. It worth noting that industry and academic prototypes make use of many different sensors to perform localization. Nevertheless, our rudimentary model serves as a base upon which to develop our understanding of image acquisition in autonomous vehicles. One of the simplified aspects of the model is the sensing device. Indeed, we adopt an ideal pinhole camera because it affords a straightforward approach to describing the behavior of image acquisition and its impact on localization. A standard, non-pinhole camera with a lens exhibits similar behavior, but is more complicated to describe mathematically. The pinhole camera we adopt for our model consists of a cube with an infinitesimally small hole in the center of one of the faces. The face opposite the hole is the focal plane of camera. The distance between the center of the focal plane and the hole is the focal length. Figure 2.1 shows the pinhole camera and illustrates how an image is projected onto the focal plane. This corresponds to the special case where the object is positioned parallel to the focal plane. We emphasize that the image is an inverted and scaled representation of the object. To simplify treatment, images on the focal plane will be described using the coordinate

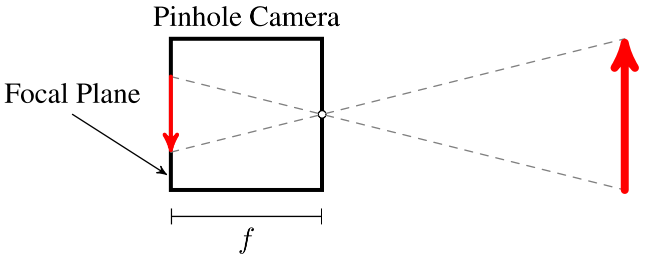


Figure 2.1: This drawing depicts a pinhole camera with focal length f. It illustrates some basic principles behind image acquisition, as they pertain to image quality.

system in Figure 2.2. An additional coordinate system will be defined to describe objects in view of the camera in the following section.

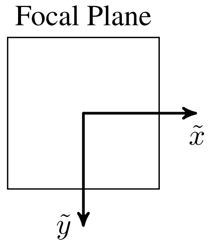


Figure 2.2: This drawing shows the coordinate system we will use to describe images on the focal plane of the camera.

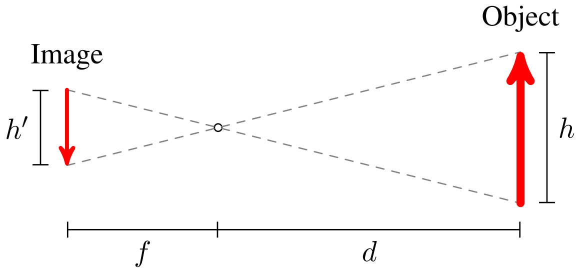


Figure 2.3: This drawing provides variable labels with which we can define the relationship between the height of the image and the height of the object.

We derive the relationship between the height of the image on the focal plane and the height of the object using similar triangles. Figure 2.3 shows the image and object from Fig. 2.1 with the body of the camera removed. The height of the object h' is shown to be

$$\frac{h'}{f} = \frac{h}{d}$$
$$h' = \frac{fh}{d}.$$

For a specific pinhole camera and object, f and h are fixed and thus h' is dependent on d, the distance between the object and the pinhole. If we take two objects with height h and place them at different distances from the pinhole, the image of the closer object will have a greater height as illustrated in Fig. 2.4. The scaling that occurs as a function of distance will be discussed further in later sections.

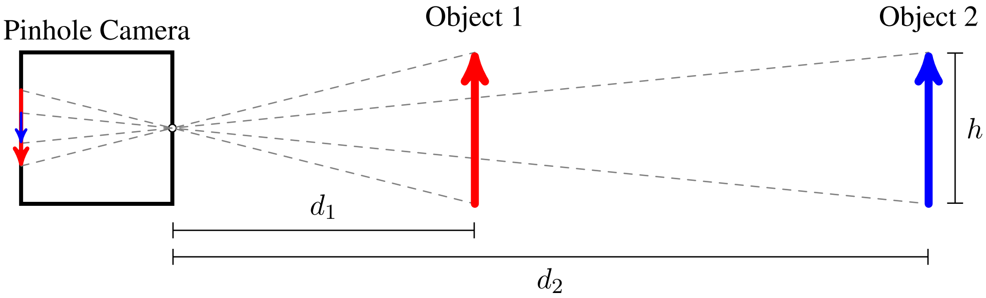


Figure 2.4: The image of Object 1 is larger than the image of Object 2 even though the objects are of equal height h.

An ideal pinhole camera is useful in our model because it does not suffer from linear distortion and has a near-infinite depth of field [22]. This allows us to simply and accurately describe the image acquisition process which in turn keeps the derivations in later sections concise.

2.1.1 Mounted Pinhole Camera on Frame Structure

With our sensing device selected, we can now build our model for image acquisition in autonomous vehicles. The information necessary for a vehicle to navigate is contained on the road surface, which we assume is planar. We place the pinhole camera on the roof of the vehicle such that both near and far sections of the road are in view. More specifically, the model contains a single pinhole camera at height h above the road surface as shown in Fig. 2.5. The camera is directed at an angle θ below the horizon line. Note the focal plane of the camera is not parallel with the road, the surface that contains the information being captured. This aspect of the data acquisition system is a differentiating characteristic of autonomous vehicles, and it has effects on signal reliability, which we will discuss in later sections.

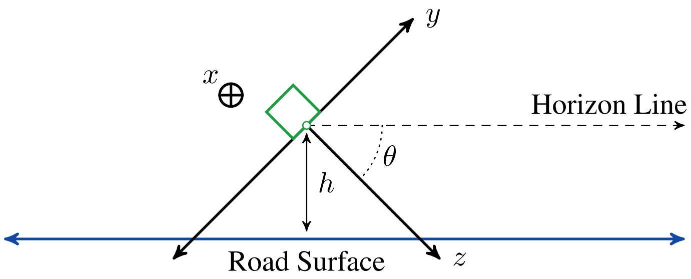


Figure 2.5: This drawing shows the pinhole camera, external coordinate system, and their orientation in relation with the planar road surface.

The model relies on two coordinate systems: one external to the camera and one internal. The external coordinate system is oriented as shown in Fig. 2.5 with its origin located at the pinhole of the camera. It is described by variables x, y, and z. Figure 2.6 depicts the orientation of the internal coordinate system relative to x, y, and z. The internal coordinate system lies on the focal plane of the camera with its origin located at $(0, 0, -f)$ in external coordinates. Its two axes, described by $\tilde{x}$ and $\tilde{y}$, are oriented such that $\tilde{x} = -x$ and $\tilde{y} = -y$. This allows us to circumvent the image inversion shown in Fig. 2.1 in our impending derivations.

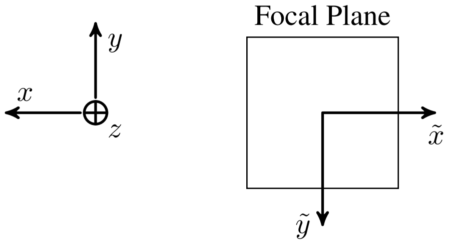


Figure 2.6: The external (left) and internal (right) coordinate axes are oriented so image inversion may be ignored in our derivations.

2.1.2 Perspective Transformation

Perspective transformations describe how different perspectives affect the way objects appear [23]. The effects of such a transformation are clearly seen in Fig. 2.7. Objects that are closer to the camera appear larger. Lines that are parallel to the focal plane of the camera are not distorted while all others are. One of the most recognizable aspects of the transformation is that parallel lines traveling away from the camera appear to converge in the distance. These effects should be very familiar to the reader as similar processes occur in human vision. The perspective transformation that objects in our model undergo when captured by the camera is described by

$$\tilde{x} = \frac{fx}{z} \qquad \tilde{y} = \frac{fy}{z}. \tag{2.1}$$

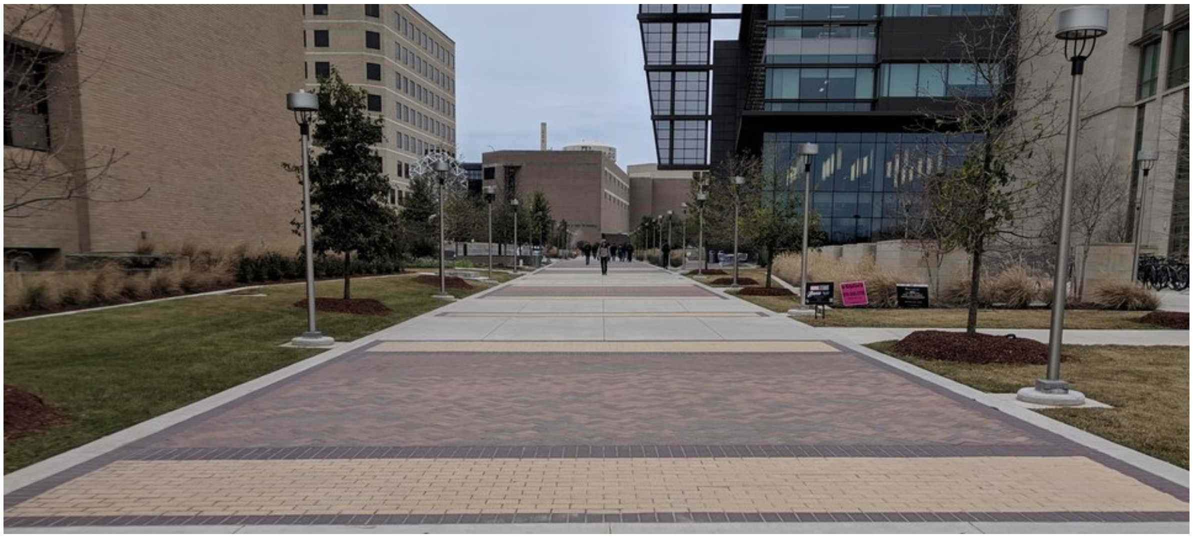


Figure 2.7: This image illustrates the affects of the perspective transformation. For example, the edges of the walkway, though parallel, appear to converge in the distance.

2.2 Focal Plane Area

In this section, we examine how features on the road are distorted as they are projected onto the focal plane of the camera. Specifically, we seek to quantify the relationship between the area of a feature on the road and its area within a captured image. To facilitate this process, we turn to a Riemann-style analysis and restrict our attention to square features obtained by placing a grid pattern on the road. Every square possesses the same area and an arbitrary grayscale value. Figure 2.8 illustrates the distortion that occurs when capturing a section of grid road. The distortion of the grid squares is not uniform throughout the image, with squares farther from the camera being projected onto smaller portions of the focal plane. This behavior has significant impacts on signal quality.

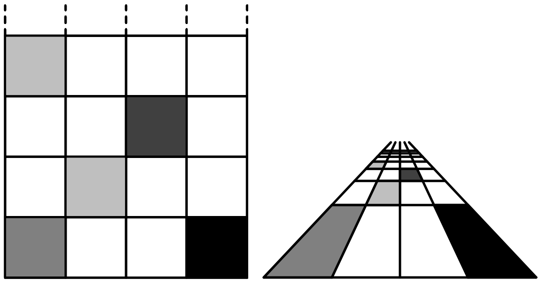


Figure 2.8: The drawing on the left is an example section of grid patterned road in front of an autonomous vehicle. When captured by the vehicle's camera its image is distorted as it is projected onto the focal plane. The image of this section of road is shown in the right drawing. Note all squares on the road have the same area but their images on the focal plane do not.

We now begin our derivation of the relationship between the area of a square feature on the road and the area of its image. We start by simplifying (2.1). The planar road in Fig. 2.5 can be described with two variables: x and z. The third component for points on the road surface becomes

$$y = z\frac{\sin\theta}{\cos\theta} - \frac{h}{\cos\theta} = z\tan\theta - h\sec\theta,$$

where θ is the angle between the horizon line and the z axis, as shown in Fig. 2.5. Using this equation we can rewrite (2.1) as functions of x and z,

$$\tilde{x} = \frac{fx}{z} \tag{2.2}$$

$$\tilde{y} = \frac{fy}{z} = \frac{f(z\tan\theta - h\sec\theta)}{z} = f\tan\theta - \frac{fh}{z}\sec\theta. \tag{2.3}$$

It is also necessary to solve (2.3) for z for later use. This is found to be

$$z = \frac{fh\sec\theta}{(f\tan\theta - \tilde{y})} = \frac{fh}{f\sin\theta - \tilde{y}\cos\theta}. \tag{2.4}$$

Recall from calculus that the absolute value of the Jacobian determinant of a transformation describes how the area around a point grows or shrinks as it undergoes the transformation. With our simplified expressions in (2.2) and (2.3), we find the Jacobian of the perspective transformation to be

$$\mathbf{J} = \begin{bmatrix} \frac{\partial\tilde{x}}{\partial x} & \frac{\partial\tilde{y}}{\partial x} \\ \frac{\partial\tilde{x}}{\partial z} & \frac{\partial\tilde{y}}{\partial z} \end{bmatrix} = \begin{bmatrix} \frac{f}{z} & 0 \\ -\frac{fx}{z^2} & -\frac{fh}{z^2}\sec\theta \end{bmatrix}. \tag{2.5}$$

The determinant of (2.5) is

$$\det(\mathbf{J}) = \begin{vmatrix} \frac{f}{z} & 0 \\ -\frac{fx}{z^2} & -\frac{fh}{z^2}\sec\theta \end{vmatrix} = -\frac{f^2h\sec\theta}{z^3}. \tag{2.6}$$

Based on this analysis, we obtain a straightforward, yet pertinent result related to road features. Specifically, this enables us to compute the area corresponding to the projection of a square on the road onto the camera's focal plane. This is accomplished in Lemma 1.

Lemma 1. *Consider rectangular region $\mathcal{R} = [x_\mathrm{l}, x_\mathrm{u}] \times [z_\mathrm{l}, z_\mathrm{u}]$ residing on the road ahead of an autonomous vehicle. The area of the projection of $\mathcal{R}$ onto the focal plane of the camera is*

$$\tilde{\mathcal{A}} = f^2h\sec\theta\frac{(x_\mathrm{u} - x_\mathrm{l})}{2}\left(\frac{1}{z_\mathrm{l}^2} - \frac{1}{z_\mathrm{u}^2}\right). \tag{2.7}$$

Proof. The area of $\tilde{\mathcal{R}}$ can be expressed as a double integral in the internal coordinate system associated with the focal plane of the camera. However, the actual computations are easier to carry using the external coordinate system. We therefore use a change of variables and leverage (2.6) to get

$$\begin{aligned} \tilde{\mathcal{A}} &= \iint_{\tilde{\mathcal{R}}} d\tilde{x} d\tilde{y} = \iint_{\mathcal{R}} |\det(\mathbf{J})| dx dz \\ &= \int_{z_{\mathrm{l}}}^{z_{\mathrm{u}}} \int_{x_{\mathrm{l}}}^{x_{\mathrm{u}}} \frac{f^2 h \sec\theta}{z^3} dx dz \\ &= f^2 h \sec\theta \frac{(x_{\mathrm{u}} - x_{\mathrm{l}})}{2} \left(\frac{1}{z_{\mathrm{l}}^2} - \frac{1}{z_{\mathrm{u}}^2} \right). \end{aligned}$$

This is the desired equation, which appears in (2.7). We emphasize that area $\tilde{\mathcal{A}}$ is proportional to the width of the rectangle. However, the distance of $\mathcal{R}$ from the autonomous vehicle is crucial in determining the effects of the transformation onto the focal plane. The footprint of the road feature decreases considerably as a function of z. □

While Lemma 1 examines rectangular features, the methodology outlined in the proof applies broadly to different shapes. Furthermore, for a small rectangular region, the infinitesimal projection onto the focal plane is well approximated by

$$\tilde{\mathcal{A}} \approx |\det(\mathbf{J})| \Delta_x \Delta_z \propto \frac{\Delta_x \Delta_z}{z^3} = \frac{\mathcal{A}}{z^3}, \tag{2.8}$$

where z is the distance of the rectangular region, along the z axis, with respect to the camera.

2.3 Non-Uniform Signal Quality

We will use the area relationship given in (2.7) to examine the signal quality in images acquired by autonomous vehicles. Not surprisingly, the nonuniform distortion of area over an image results in nonuniform signal quality. We derive a mathematical expression for the signal quality in this section.

Suppose there is a function $b(x, z)$ which, at a moment in time, acquires the amplitude present at point (x, z) on the road. Then, there is a similar function $\tilde{b}(\tilde{x}, \tilde{y})$ that represents the amplitude at point $(\tilde{x}, \tilde{y})$ at the same moment in time, where $(\tilde{x}, \tilde{y})$ is the projection of (x, z) on the focal plane. We use (2.2) and (2.4) to express this function as

$$\tilde{b}(\tilde{x}, \tilde{y}) = b\left(\frac{h\tilde{x}}{f\sin\theta - \tilde{y}\cos\theta}, \frac{fh}{f\sin\theta - \tilde{y}\cos\theta}\right).$$

The camera sensor, located on the focal plane, acquires a signal at point $(\tilde{x}, \tilde{y})$ which is a combination of amplitude and noise. This is represented by $\tilde{b}(\tilde{x}, \tilde{y}) + N(\tilde{x}, \tilde{y})$, where N has power spectral density N_0. Thus, the signal acquired over a region $\tilde{\mathcal{R}}$ of the sensor is found to be

$$S = \underbrace{\iint_{\tilde{\mathcal{R}}} \tilde{b}(\tilde{x}, \tilde{y}) d\tilde{x} d\tilde{y}}_{\text{amplitude}} + \underbrace{\iint_{\tilde{\mathcal{R}}} N(\tilde{x}, \tilde{y}) d\tilde{x} d\tilde{y}}_{\text{noise}}.$$

We model the noise introduced by the camera sensor using an additive zero-mean, white Gaussian random process. This is appropriate assuming the sensor is working in its linear or non-saturation region. Within region $\tilde{\mathcal{R}}$ the variance of the noise is proportional to $\tilde{\mathcal{A}}$, the area of region $\tilde{\mathcal{R}}$ on the focal plane. The power spectral density N_0 is unknown as it depends on many factors, but constant over the entire sensor. Thus, $\sigma^2 = \tilde{\mathcal{A}}N_0$.

2.3.1 Regions of Uniform Amplitude

Suppose all points in a rectangular region on the road surface have the same amplitude. Described mathematically, $b(x, z) = a$ for all $(x, z) \in \mathcal{R}$, where $\mathcal{R} = [x_{\mathrm{l}}, x_{\mathrm{u}}] \times [z_{\mathrm{l}}, z_{\mathrm{u}}]$. This restriction is made without much loss of generality as the regions can be made arbitrarily small. We can compute the effective SNR of $\tilde{\mathcal{R}}$, the image of region $\mathcal{R}$ on the focal plane, by the following steps.

The energy associated with the signal component over $\tilde{\mathcal{R}}$ is

$$\left(\iint_{\tilde{\mathcal{R}}} \tilde{b}(\tilde{x}, \tilde{y}) d\tilde{x} d\tilde{y}\right)^2 = \left(\iint_{\tilde{\mathcal{R}}} a d\tilde{x} d\tilde{y}\right)^2 = a^2\tilde{\mathcal{A}}^2, \tag{2.9}$$

where $\tilde{\mathcal{A}}$ is given in (2.7). As described above, the noise introduced by the camera sensor is zero-mean Gaussian with variance $\tilde{\mathcal{A}}N_0$. We assume this noise is independent of $\tilde{b}(\tilde{x},\tilde{y})$. Thus, the effective SNR corresponding to region $\mathcal{R}$ on the road becomes

$$\mathrm{SNR} = \frac{a^2\tilde{\mathcal{A}}^2}{\tilde{\mathcal{A}}N_0} = \frac{a^2\tilde{\mathcal{A}}}{N_0} = \frac{a^2}{N_0}\frac{f^2 h}{\cos\theta}\frac{(x_{\mathrm{u}} - x_{\mathrm{l}})}{2}\left(\frac{1}{z_{\mathrm{l}}^2} - \frac{1}{z_{\mathrm{u}}^2}\right). \tag{2.10}$$

Moreover, owing to (2.8), we gather that, for a small region $\mathcal{R}$, the SNR is well-approximated by

$$\mathrm{SNR} \approx \frac{a^2 f^2 h}{N_0\cos\theta}\frac{\Delta_x\Delta_z}{z^3},$$

where z is any point in $[z_{\mathrm{l}}, z_{\mathrm{u}}]$.

The SNR depends on several factors, but most importantly, it decreases heavily as a function of z. As such, road features far from a vehicle provide less reliable data than features up close. This behavior, not currently accounted for in existing autonomous systems, represents an opportunity to improve localization algorithms.

2.3.2 Examining a Grid Patterned Road

Suppose there is a grid road, as shown in Fig. 2.8, where each square has a uniform, randomly assigned amplitude that is independent of all other squares. As an autonomous vehicle drives along, it captures sections of the road with a pinhole camera. We examine the localization task for a single captured section of the road.

As discussed in Section 1.1, we restrict the localization task to matching the current captured image with a piece of the pre-built map. The section of road captured by the camera is a collection of distorted squares with SNR values as defined by (2.10). We represent these squares as a vector of amplitudes $\mathbf{a} = (a_1, \ldots, a_m)$. Thus, the observation of these squares becomes

$$\mathbf{v} = \mathbf{a} + \mathbf{n},$$

where $\mathbf{n} = (n_1, \ldots, n_m)$ is a Gaussian noise vector. Element a_k is the amplitude corresponding to region $\mathcal{R}_k$ on the road surface. N_k is the noise introduced by the camera sensor over the region

$\tilde{\mathcal{R}}_k$, the projection of $\mathcal{R}_k$ onto the focal plane. The elements of $\mathbf{n}$ are independent because none of the $\tilde{\mathcal{R}}_k$ regions overlap. Thus, element N_k has zero mean and variance

$$\sigma_k^2 = \frac{N_0}{\tilde{\mathcal{A}}_k}, \tag{2.11}$$

and the effective SNR corresponding to region $\mathcal{R}_k$ is

$$\mathrm{SNR}_k = \frac{a_k^2 \tilde{\mathcal{A}}_k}{N_0}. \tag{2.12}$$

This formulation is the same as (2.10).

2.3.3 Localization Using an Inner Product

Following directly from the previous section we examine utilizing an inner product to perform localization. The task is to match our observation vector $\mathbf{v}$ with a section of the global map denoted by $\hat{\mathbf{u}}$. The global map is assumed to be a perfect representation of the environment, in this case a grid road. Thus, each section of the map $\hat{\mathbf{u}}$ is a vector of amplitudes with length m, that is, the length of $\mathbf{v}$. The section of map that matches with $\mathbf{v}$ is clearly $\hat{\mathbf{u}} = \mathbf{a}$. However, since $\mathbf{a}$ is obscured by noise the solution is not that simple. To localize $\mathbf{v}$ we must select multiple candidate sections of the map and find the most likely match among our choices. This becomes a standard Gaussian classification problem.

The likelihood for candidate map section $\hat{\mathbf{u}}$ is found to be

$$\begin{aligned}
\mathcal{L}\left(\hat{\mathbf{u}}|\mathbf{v}\right) &= \frac{1}{\sqrt{(2\pi)^m \prod_k \sigma_k^2}} \exp\left(-\frac{1}{2}\sum_k \frac{\left(\hat{u}_k - v_k\right)^2}{\sigma_k^2}\right) \\
&= \frac{1}{\sqrt{(2\pi)^m \prod_k \sigma_k^2}} \exp\left(-\frac{1}{2N_0}\sum_k \tilde{\mathcal{A}}_k \left(\hat{u}_k - v_k\right)^2\right) \\
&= \frac{1}{\sqrt{(2\pi)^m \prod_k \sigma_k^2}} \exp\left(-\frac{f^2 h \sec\theta}{N_0}\sum_k \frac{\left(x_{\mathrm{u},k} - x_{\mathrm{l},k}\right)}{2}\left(\frac{1}{z_{\mathrm{l},k}^2} - \frac{1}{z_{\mathrm{u},k}^2}\right)\left(\hat{u}_k - v_k\right)^2\right) \\
&= \frac{1}{\sqrt{(2\pi)^m \prod_k \sigma_k^2}} \exp\left(-\frac{f^2 h \sec\theta}{2N_0}\sum_k \mathbf{G}_{k,k}\left(\hat{u}_k - v_k\right)^2\right).
\end{aligned}$$

The Gramian matrix $\mathbf{G}$ is positive-definite and diagonal. Its entries, as seen above, are given by

$$\mathbf{G}_{k,k} = \frac{(x_{\mathrm{u},k} - x_{\mathrm{l},k})}{2} \left(\frac{1}{z_{\mathrm{l},k}^2} - \frac{1}{z_{\mathrm{u},k}^2} \right) = \frac{\tilde{\mathcal{A}}_k}{f^2 h \sec\theta}. \tag{2.13}$$

The maximum likelihood (ML) decision rule for this classification task can be expressed as

$$\mathbf{u}_{\mathrm{ML}}^* (\mathbf{v}) = \arg\min_{\hat{\mathbf{u}}} \|\mathbf{v} - \hat{\mathbf{u}}\|_{\mathbf{G}} , \tag{2.14}$$

where "$\| \cdot \|_{\mathbf{G}}$ is the norm induced by the generalized inner product $\langle \mathbf{w}_1 | \mathbf{w}_2 \rangle_{\mathbf{G}} = \mathbf{w}_2^{\mathrm{T}} \mathbf{G} \mathbf{w}_1$" [24]. This can be viewed as an instance of maximal ratio combining.

In our grid road model with uniform amplitude squares the Gramian matrix does not depend on the camera height h, camera angle θ, focal length f, power spectral density N_0, or even the amplitude values of the squares. Changing these parameters may impact localization performance, but will not change the structure of the ML classifier in (2.14). In practice, a Bayesian approach is often used for localization where prior information of the vehicle is used to estimate its current position. This can be done using an extended Kalman filter (EKF), for example. The location estimate is then confirmed or altered based on where the current image matches with the global map. Even in this instance, the structure of the Gramian matrix and ML classifier remains the same. Though the location estimate may decrease the search space within the global map, it does not change the image matching task or the nonuniform noise we account for in our classifier.

For finely quantized images, the diagonal weights of the Gramian matrix approach values proportional to the absolute value of the Jacobian determinant evaluated at the corresponding coordinate points. Mathematically, suppose that we center intervals $[x_{\mathrm{l},k}, x_{\mathrm{u},k}]$ and $[z_{\mathrm{l},k}, z_{\mathrm{u},k}]$ around

(x_k, z_k). Then, we get

$$\begin{aligned}
\mathbf{G}_{k,k} &= \frac{(x_{\mathrm{u},k} - x_{\mathrm{l},k})}{2}\left(\frac{1}{z_{\mathrm{l},k}^2} - \frac{1}{z_{\mathrm{u},k}^2}\right) = \frac{(x_{\mathrm{u},k} - x_{\mathrm{l},k})}{2}\left(\frac{z_{\mathrm{u},k}^2}{z_{\mathrm{l},k}^2 z_{\mathrm{u},k}^2} - \frac{z_{\mathrm{l},k}^2}{z_{\mathrm{l},k}^2 z_{\mathrm{u},k}^2}\right) \\
&= \frac{(x_{\mathrm{u},k} - x_{\mathrm{l},k})}{2}\left(\frac{z_{\mathrm{u},k}^2 - z_{\mathrm{l},k} z_{\mathrm{u},k} + z_{\mathrm{l},k} z_{\mathrm{u},k} - z_{\mathrm{l},k}^2}{z_{\mathrm{l},k}^2 z_{\mathrm{u},k}^2}\right) \\
&= \frac{(x_{\mathrm{u},k} - x_{\mathrm{l},k})\,(z_{\mathrm{u},k} - z_{\mathrm{l},k})}{2}\left(\frac{z_{\mathrm{u},k} + z_{\mathrm{l},k}}{z_{\mathrm{l},k}^2 z_{\mathrm{u},k}^2}\right) \\
&= \frac{\Delta_x \Delta_z}{2}\left(\frac{z_{\mathrm{u},k} + z_{\mathrm{l},k}}{z_{\mathrm{l},k}^2 z_{\mathrm{u},k}^2}\right) \approx \frac{\Delta_x \Delta_z}{z_k^3} \propto |\det(\mathbf{J}(z_k))|\Delta_x \Delta_z.
\end{aligned} \tag{2.15}$$

Thus, for finely quantized images, the collected pixel values should be weighted by the Jacobian determinant $|\det(\mathbf{J})|$ evaluated at distance z.

The key point of this section is that the ML classifier in (2.14) should be used for image matching in AV localization. Using the Euclidean norm or standard inner product implicitly assumes uniform noise over the entire image. We have shown through our derivations that this assumption is not valid. Our classifier, due to its use of $\|\cdot\|_{\mathbf{G}}$, accounts for the unequal noise distribution introduced by the camera and its orientation on the vehicle.

2.3.4 Localization Using Normalized Mutual Information

We now turn to normalized mutual information (NMI) as a technique for localization. Again we must match an observation vector $\mathbf{v}$ with a section of the global map $\hat{\mathbf{u}}$. For a particular observation $\mathbf{v}$, we select multiple candidate sections of the map, denoted by $(\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_t)$. The candidate section that results in the highest NMI value with $\mathbf{v}$ is declared a match. Mathematically, this decision rule is expressed as

$$\mathbf{u}^*_{\mathrm{NMI}}(\mathbf{v}) = \arg\max_{\hat{\mathbf{u}}} \left[\mathrm{NMI}(\hat{\mathbf{u}}, \mathbf{v})\right]. \tag{2.16}$$

NMI has advantageous properties that make it an effective technique for localization. However, it does not account for the unequal distribution of noise present in our images. We look to modify NMI to account for this property and start by examining the distribution of the observation vector.

The joint PDF of $\mathbf{v}$ is

$$f_{\mathbf{v}}(\mathbf{v}|\mathbf{a}) = \frac{1}{\sqrt{(2\pi)^m \prod_k \sigma_k^2}} \exp\left(-\frac{1}{2}\sum_k \frac{(v_k - a_k)^2}{\sigma_k^2}\right).$$

Thus, the likelihood function for $\mathbf{a}$ is shown to be

$$\mathcal{L}\left(\mathbf{a}|\mathbf{v}\right) = \frac{1}{\sqrt{(2\pi)^m \prod_k \sigma_k^2}} \exp\left(-\frac{1}{2}\sum_k \frac{(a_k - v_k)^2}{\sigma_k^2}\right).$$

The likelihood for the k^{th} grid square reduces to

$$\mathcal{L}(a_k|v_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{(a_k - v_k)^2}{2\sigma_k^2}\right), \tag{2.17}$$

where σ_k^2 is given by (2.11).

We can account for the uncertainty in $\mathbf{v}$ through modifications to NMI. For each value in $\mathbf{v}$ we spread its contribution to the joint distribution over possible amplitude values in accordance with (2.17). We call our technique enhanced normalized mutual information (ENMI) and will describe it in the following chapter. Our decision rule is slightly modified to include this

$$\mathbf{u}^*_{\mathrm{ENMI}}\left(\mathbf{v}\right) = \arg\max_{\hat{\mathbf{u}}}\left[\mathrm{ENMI}\left(\hat{\mathbf{u}}, \mathbf{v}\right)\right]. \tag{2.18}$$

2.3.4.1 An Overview of Mutual Information

We provide a mathematical description of mutual information (MI) to complete our discussion on NMI for localization. As described in Section 1.2, MI measures the quantity of shared information between two random variables. Given the random variables A and B defined over $\mathcal{A}$ and $\mathcal{B}$, respectively, the MI between them, denoted by I, is given by

$$\mathrm{I}(A;B) = \sum_{A\in\mathcal{A}}\sum_{B\in\mathcal{B}} p_{(A,B)}(a,b) log\left(\frac{p_{(A,B)}(a,b)}{p_A(a)p_B(b)}\right). \tag{2.19}$$

Note p_A and p_B are the marginal probability mass functions of A and B; $p_{(A,B)}$ is the joint probability mass function of A and B. Alternatively, MI can be expressed as

$$\mathrm{I}(A;B) = H[A] + H[B] - H[A,B], \tag{2.20}$$

where $H[A]$ and $H[B]$ are the (information) entropy of A and B, respectively, and $H[A,B]$ is their joint entropy. Entropy quantifies the average amount of information contained in a random variable. The entropy of random variable A defined over $\mathcal{A}$ is given by

$$H[A] = -\sum_{A\in\mathcal{A}} p_A(a) log\left(p_A(a)\right).$$

The joint entropy of random variables A and B defined over $\mathcal{A}$ and $\mathcal{B}$, respectively, is given by

$$H[A,B] = -\sum_{A\in\mathcal{A}}\sum_{B\in\mathcal{B}} p_{(A,B)}(a,b) log(p_{(A,B)}(a,b)).$$

The equivalence between 2.19 and 2.20 is shown by

$$\begin{aligned}
\mathrm{I}(A;B) &= H[A] + H[B] - H[A,B] \\
&= -\sum_{A\in\mathcal{A}} p_A(a) log(p_A(a)) - \sum_{B\in\mathcal{B}} p_B(b) log(p_B(b)) + \sum_{A\in\mathcal{A}}\sum_{B\in\mathcal{B}} p_{(A,B)}(a,b) log(p_{(A,B)}(a,b)) \\
&= -\sum_{A\in\mathcal{A}}\sum_{B\in\mathcal{B}} p_{(A,B)}(a,b)\left[log(p_A(b)) + log(p_B(b))\right] + \sum_{A\in\mathcal{A}}\sum_{B\in\mathcal{B}} p_{(A,B)}(a,b) log(p_{(A,B)}(a,b)) \\
&= -\sum_{A\in\mathcal{A}}\sum_{B\in\mathcal{B}} p_{(A,B)}(a,b) log(p_A(b)p_B(b)) + \sum_{A\in\mathcal{A}}\sum_{B\in\mathcal{B}} p_{(A,B)}(a,b) log(p_{(A,B)}(a,b)) \\
&= \sum_{A\in\mathcal{A}}\sum_{B\in\mathcal{B}} p_{(A,B)}(a,b) log\left(\frac{p_{(A,B)}(a,b)}{p_A(a)p_B(b)}\right).
\end{aligned}$$

In this paper, we use a normalized measure of MI proposed by Studholme et al. [18] to calculate

NMI values. This measure is defined as

$$\mathrm{NMI}[A,B] = \frac{H[A]+H[B]}{H[A,B]} \tag{2.21}$$

and is a normalized version of 2.20 equivalent to

$$\begin{aligned}
\mathrm{NMI}[A,B] &= \frac{H[A]+H[B]}{H[A,B]} \\
&= \frac{H[A]+H[B]}{H[A,B]} - \frac{H[A,B]}{H[A,B]} + 1 \\
&= \frac{H[A]+H[B]-H[A,B]}{H[A,B]} + 1 \\
&= \frac{\mathrm{I}(A;B)}{H[A,B]} + 1.
\end{aligned}$$

3. LOCALIZATION IMPROVEMENTS

3.1 Weighted Inner Product

An inner product provides a clear entry point for our noise model. This technique, though not used for localization in practice, is easily described and evaluated mathematically.

3.1.1 Performance Assessment

Below, we evaluate the performance of our classifier in (2.14). For simplicity, our treatment assumes a grid road with uniform amplitude squares as presented above. Additionally, we restrict the amplitude of the road squares to $\pm a$ to allow a straightforward assessment of the technique. Though this restriction is only a first-order approximation, it has similarities with how roads are designed today, using dark pavement and light road markings. The amplitude values acquired from these parts of the road will have a gap, and when this gap is centered on the value 0, the road should generally follow the $\pm a$ structure. With these assumptions in place, we can assess the classifier mathematically.

Using (2.14) we know the true location, denoted by $\mathbf{u}^*$, is selected as a match whenever

$$\|\mathbf{v} - \mathbf{u}^*\|_{\mathbf{G}}^2 \leq \|\mathbf{v} - \hat{\mathbf{u}}\|_{\mathbf{G}}^2 \quad \forall\, \hat{\mathbf{u}} \neq \mathbf{u}^*. \tag{3.1}$$

Given a known true location and an alternative location, we can rewrite this condition as

$$\|\mathbf{v} - \mathbf{u}^*\|_{\mathbf{G}}^2 - \|\mathbf{v} - \hat{\mathbf{u}}\|_{\mathbf{G}}^2 < 0. \tag{3.2}$$

Recall the square of the induced norm of $\mathbf{v} - \mathbf{u}$ can be expanded as follows,

$$\|\mathbf{v} - \mathbf{u}\|_{\mathbf{G}}^2 = \langle \mathbf{v} - \mathbf{u} | \mathbf{v} - \mathbf{u} \rangle_{\mathbf{G}} = \|\mathbf{v}\|_{\mathbf{G}}^2 - 2 \langle \mathbf{u} | \mathbf{v} \rangle_{\mathbf{G}} + \|\mathbf{u}\|_{\mathbf{G}}^2 .$$

Thus, the condition in (3.2) reduces to

$$\begin{aligned}0 &> \|\mathbf{v}\|_{\mathbf{G}}^2 - 2\left\langle \mathbf{u}^*|\mathbf{v}\right\rangle_{\mathbf{G}} + \|\mathbf{u}^*\|_{\mathbf{G}}^2 - \|\mathbf{v}\|_{\mathbf{G}}^2 + 2\left\langle \hat{\mathbf{u}}|\mathbf{v}\right\rangle_{\mathbf{G}} - \|\hat{\mathbf{u}}\|_{\mathbf{G}}^2 \\ &= 2\left\langle \hat{\mathbf{u}}|\mathbf{v}\right\rangle_{\mathbf{G}} - 2\left\langle \mathbf{u}^*|\mathbf{v}\right\rangle_{\mathbf{G}}\end{aligned}$$

or, equivalently,

$$0 < \left\langle \mathbf{u}^* - \hat{\mathbf{u}}|\mathbf{v}\right\rangle_{\mathbf{G}} = \left\langle \mathbf{u}^* - \hat{\mathbf{u}}|\mathbf{u}^*\right\rangle_{\mathbf{G}} + \left\langle \mathbf{u}^* - \hat{\mathbf{u}}|\mathbf{n}\right\rangle_{\mathbf{G}}.$$

We emphasize that $\|\mathbf{u}^*\|_{\mathbf{G}}^2 = \|\hat{\mathbf{u}}\|_{\mathbf{G}}^2$ due to our $\pm a$ amplitude restriction. The first component is a known constant and can be expressed as

$$\left\langle \mathbf{u}^* - \hat{\mathbf{u}}|\mathbf{u}^*\right\rangle_{\mathbf{G}} = \|\mathbf{u}^*\|_{\mathbf{G}}^2 - \left\langle \hat{\mathbf{u}}|\mathbf{u}^*\right\rangle_{\mathbf{G}} = ma^2 - \left\langle \hat{\mathbf{u}}|\mathbf{u}^*\right\rangle_{\mathbf{G}},$$

where m is the length of the true location vector. The second component, later referred to as the noise component, is a zero-mean Gaussian random variable [24]. It can be expressed as

$$\left\langle \mathbf{u}^* - \hat{\mathbf{u}}|\mathbf{n}\right\rangle_{\mathbf{G}} = \sum_k (u_k^* - \hat{u}_k)\mathbf{G}_{k,k} n_k.$$

Using (2.13) we find the variance of the noise component to be

$$\begin{aligned}\sum_k (u_k^* - \hat{u}_k)^2 \mathbf{G}_{k,k}^2 \sigma_k^2 &= \sum_k \frac{(u_k^* - \hat{u}_k)^2 \mathbf{G}_{k,k}^2 N_0}{\tilde{\mathcal{A}}_k} \\ &= \frac{N_0}{f^2 h \sec\theta} \sum_k (u_k^* - \hat{u}_k)^2 \mathbf{G}_{k,k}.\end{aligned}$$

Given this, we can show the probability of error given one alternative to be

$$1 - \boldsymbol{\Phi}\left(\sqrt{\frac{f^2 h \sec\theta}{N_0}} \frac{\left\langle \mathbf{u}^* - \hat{\mathbf{u}}|\mathbf{u}^*\right\rangle_{\mathbf{G}}}{\|\mathbf{u}^* - \hat{\mathbf{u}}\|_{\mathbf{G}}}\right), \tag{3.3}$$

where $\boldsymbol{\Phi}(\cdot)$ is the CDF of the normal Gaussian distribution. In comparison, if we use the standard inner product for classification, performance deteriorates and the probability of error given one alternative becomes

$$1 - \mathbf{\Phi}\left(\sqrt{\frac{f^2 h \sec\theta}{N_0}}\frac{\langle \mathbf{u}^* - \hat{\mathbf{u}} | \mathbf{u}^* \rangle}{\|\mathbf{u}^* - \hat{\mathbf{u}}\|_{\mathbf{G}^{-1}}}\right). \tag{3.4}$$

3.1.2 Simulated Performance

We evaluate the performance of the standard and generalized inner product using numerical simulations. Performance is measured using the probability of error as defined in the previous section. We randomly generate true and alternative location vectors, $\mathbf{u}^*$ and $\hat{\mathbf{u}}$, with amplitudes $\pm a$. These vectors simulate captured images similar to Fig. 3.1.

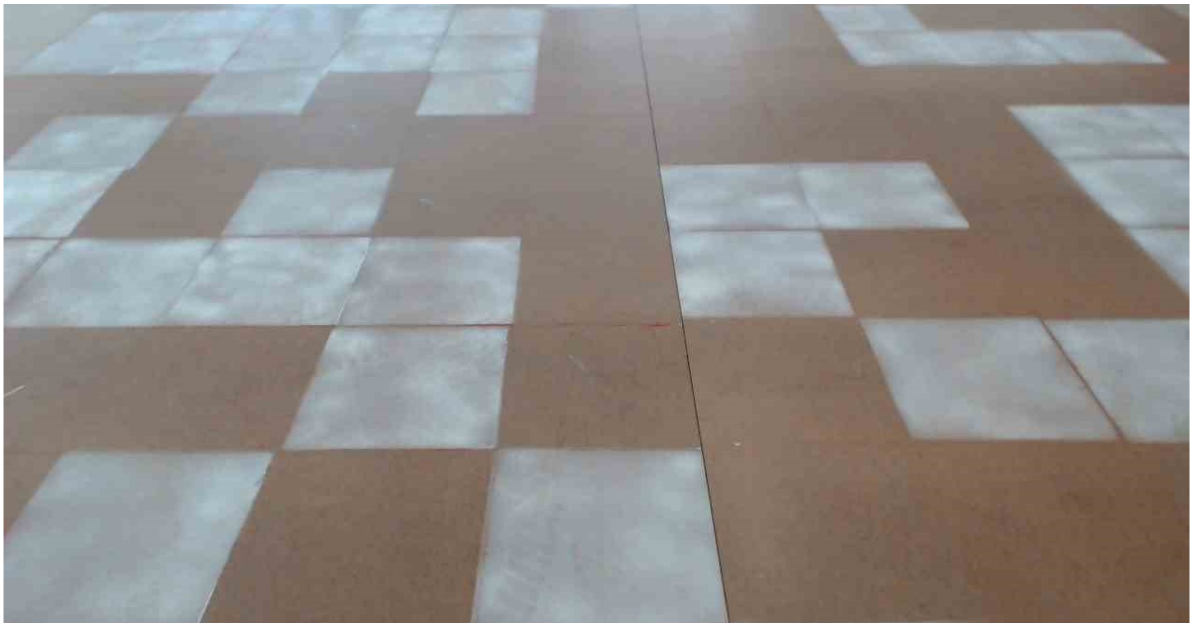


Figure 3.1: This is a sample image of part of a grid road.

The listed parameters yield images of 66 whole squares. Once rectified, these images can be represented as vectors of $\pm a$ amplitude squares and used in our simulations. Figure 3.2 shows a rectified representation of the example image in Fig. 3.1. Note the squares collectively form a trapezoid once rectified.

We vary the amplitude a from 10 to 0.1 which decreases the SNR in accordance with (2.10).

Parameters
f = 0.0367 cm
θ = 35.9020°
h = 58.3095 cm
Side Length = 20 cm
Vertical View: 39.3°
Horizontal View: 70.5°
10,000 trials per point

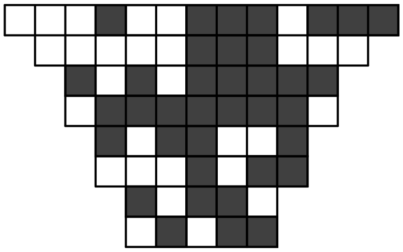


Figure 3.2: This grid shows an example rectified observation of 66 whole squares. The dark and light squares represent amplitudes of $-a$ and $+a$, respectively.

Ten thousand $\mathbf{u}^*$ and $\hat{\mathbf{u}}$ vectors are randomly generated for each a value. The probabilities of error are calculated using (3.3) and (3.4). Figure 3.3 shows the results of the simulation.

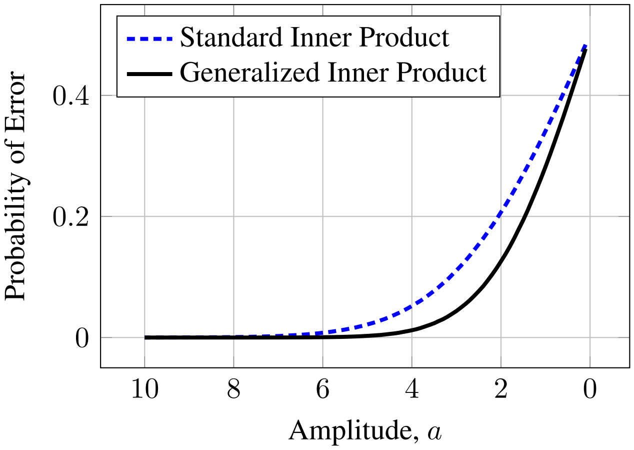


Figure 3.3: This figure showcases the performance gains of the generalized inner product. Note SNR is proportional to a^2.

3.2 Enhanced Normalized Mutual Information

Before we can describe our modifications to NMI, we must review the technique and its use in localization. The task is to match an observation $\mathbf{v}$ with a section of the global map $\hat{\mathbf{u}}$. We select a matching section of the global map, denoted by $\mathbf{u}^*$, using the classifier in (2.16). The location of the vehicle is determined based on the metadata of $\mathbf{u}^*$.

3.2.1 Calculating NMI

Before we introduce modifications we must examine how the NMI of two images is calculated. We refer to a captured image as the current image taken by an AV. This image is represented as an observation vector $\mathbf{v}$. Additionally, we refer to a map section as a part of the global map of the same size and scale as the captured image. We represent each map section as a candidate vector $\hat{\mathbf{u}}$. As an illustration, let a captured image and map section be 3 by 3 images as shown in Fig. 3.4. These can represent sections of 9 squares from a grid road, or perhaps more realistically, images of 3 pixels by 3 pixels. The values in the images can be thought of as gray-scale, amplitude, or intensity values. They are kept small simply to keep the joint distribution small as its size depends on the range of possible values and the bin size. In Fig. 3.4, the joint distribution has bins for each value or a bin size of 1. This would not be the case in practice, but it is done to make the figure simple and clear.

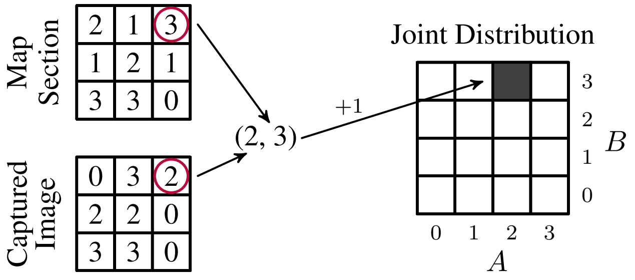


Figure 3.4: This diagram shows how the top-right pair of values adds to the empirical joint distribution. A corresponds to the values in the captured image; B corresponds to the values in the map section. The range of possible values is limited for simplicity.

To compute the NMI between these two images we first couple their values based on position. Next, we bin each pair of values to create a joint distribution. Fig. 3.4 illustrates this process for the top-right position in the images. After the joint distribution is complete, we can calculate the NMI value between the captured image and map section. Let (A, B) be a random vector drawn from the empirical joint PMF of the image values. A is drawn from the distribution of values in the captured image; B is drawn from the distribution of values in the map section. We then use (2.21) to calculate the NMI value. We repeat this process for multiple map sections and localize the captured image using (2.16).

3.2.2 Likelihood-Based Joint Distribution

We modify the procedure described above to account for uncertainty and the unequal noise distribution present in captured images. The process begins the same by coupling values based on their position in the images. However, instead of binning each pair as before, we employ the likelihood function detailed in (2.17) to generate a maximum a posteriori probability over possible captured image values [25]. This effectively generates multiple value pairs over which the contribution of the original pair is distributed. The weight of a single value pair could be spread over multiple bins in the joint distribution depending on its effective SNR and the bin size. An example of this is shown in Fig. 3.5. Note that we only spread the weight of a value pair over the axis in the joint distribution corresponding to the captured image. A similar operation can be performed on the map section values, but since we assume a noiseless global map, this is not necessary in the current context. Once the modified joint distribution is generated, we calculate the ENMI value using the standard equation given in (2.21). We repeat this process for multiple map sections and localize the captured image using (2.18).

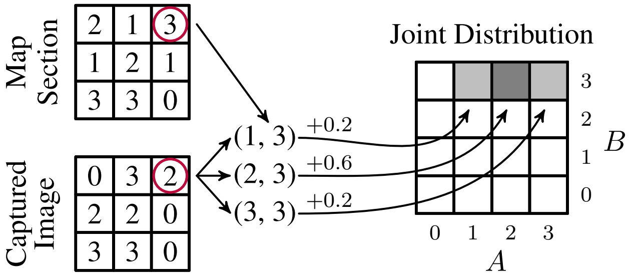


Figure 3.5: This diagram illustrates how the contribution of the top-right pair of values is distributed between multiple bins to reflect the uncertainty in the captured image value. The extent to which the contribution is spread depends on the effective SNR, calculated using (2.10), present at the top-right position in the captured image. Note the weight is spread over the A axis only.

3.2.3 Simulated Performance

We evaluate the performance of ENMI and NMI for image matching using numerical simulations. Our performance metric is the probability of error. We follow a similar procedure to the one employed to evaluate the generalized inner product in Section 3.1.2. We generate images of 66 squares using the same camera parameters as before. However, the squares in these images have amplitude values drawn from a Gaussian distribution with $\mu = 128$ and $\sigma = 32$. Our evaluation of ENMI and NMI does not require a $\pm a$ amplitude restriction. An example rectified image is shown in Fig. 3.6.

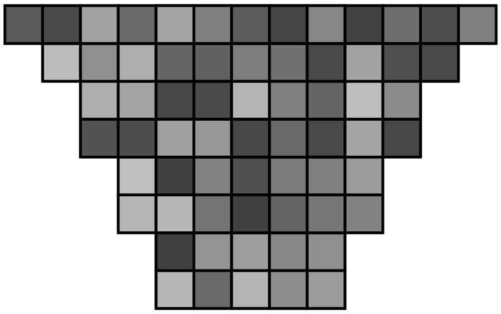


Figure 3.6: This grid of 66 squares is an example rectified image.

We change N_0, the power spectral density, to demonstrate the behavior of each technique as the noise increases. Ten thousand trials are performed per N_0 value. For each trial, we generate a true location vector $\mathbf{u}^*$ and alternate location vector $\hat{\mathbf{u}}$. Then a noisy image vector $\mathbf{v}$ is calculated using

$$\mathbf{v} = \mathbf{a} + \mathbf{n} = \mathbf{u}^* + \mathbf{n},$$

where $\mathbf{n}$ is additive zero-mean Gaussian noise with variance $N_0/\tilde{\mathcal{A}}_k$. This gives each square in $\mathbf{v}$ an effective SNR equal to (2.12). An error is recorded when an alternate location vector is selected as a match. This occurs for NMI when

$$\mathrm{NMI}[\mathbf{v}, \hat{\mathbf{u}}] \geq \mathrm{NMI}[\mathbf{v}, \mathbf{u}^*]$$

and ENMI when

$$\mathrm{ENMI}[\mathbf{v}, \hat{\mathbf{u}}] \geq \mathrm{ENMI}[\mathbf{v}, \mathbf{u}^*].$$

Figure 3.7 shows the results of our simulations. As the effective SNR decreases the probability of error increases for both techniques, however, ENMI is more resilient to increased noise.

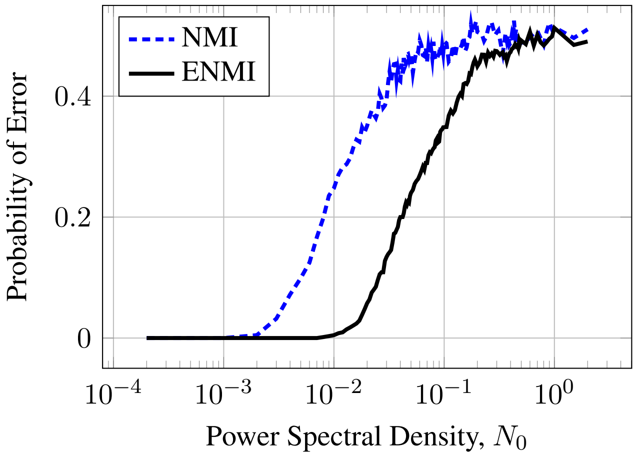


Figure 3.7: This figure illustrates the increased robustness to noise ENMI provides over the standard NMI technique.

3.3 ENMI Performance Evaluation

We evaluate the performance of ENMI due to its practical application in state of the art autonomous platforms. Our first test is done in the Unity game engine followed by offline localization testing on real-world AV data provided by Ford Autonomous Vehicles LLC.

3.3.1 Unity Simulation

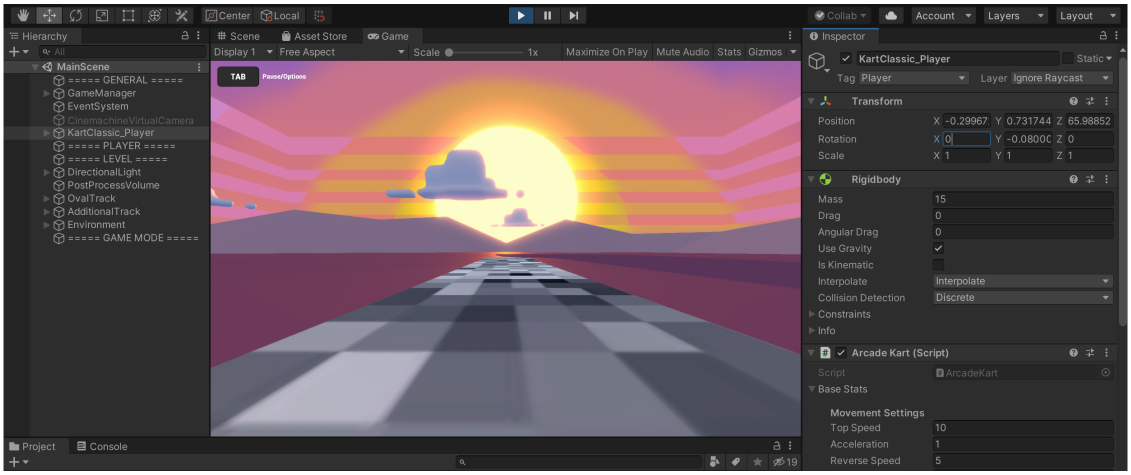
Unity is a popular game development tool that allows users to build games easily and with a high level of flexibility. The karting microgame from Unity provides a reasonable starting point for localization testing. This game includes a character on a cart that the user drives around an oval track. The Unity development hub provides the location of the kart which we will use as our ground truth. We built a simple track and patterned it with a global map similar to the grid road in Fig. 3.6. Each square's color is drawn from a Gaussian distribution with $\mu = 128$ and $\sigma = 32$. The user's viewpoint of the game is through a simulated camera positioned above the cart, 1.25 meters above the planar road. The camera is directed at the horizon. Table 3.1 includes the parameters used in the Unity simulation.

Unity Parameters
f = 8.097338 mm
θ = 0°
h = 1.25 m
Side Length = 1.67 m
Vertical View = 60°
Horizontal View = 107.16°
N_0 = 5×10^{-8} m^{-2}
Number of Bins = 16

Table 3.1: This table states the parameters used in the Unity simulation.

A screenshot of the Unity development hub is included in Fig. 3.8. We collected 100 images driving along the track with which to compare the performance of NMI and ENMI. The localization process for one image is as follows. First, a portion of the game view is cropped and transformed into the top-down perspective of the global map. This portion is a 6.67 meter wide and 35 meter tall section of road. Figure 3.9 shows the cropped portion taken from the image in Fig. 3.8 and its transformation. Next, the ground truth position is taken from Unity and used to generate a 0.5 meter by 0.5 meter search space that includes the true position. This is done in the absence of an

EKF to reduce the search space. Finally, the NMI and ENMI scores are calculated for each map section in the search space, the matches for each technique are determined, and the longitudinal and lateral errors are recorded. This process is repeated for each of the 100 images.



☑ KartClassic_Player □ Static

Tag Player Layer Ignore Raycast

Transform

Position	X	-0.29967	Y	0.731744	Z	65.98852
Rotation	X	0	Y	-0.0800C	Z	0
Scale	X	1	Y	1	Z	1

Rigidbody

Mass	15
Drag	0
Angular Drag	0
Use Gravity	☑
Is Kinematic	□
Interpolate	Interpolate
Collision Detection	Discrete

Constraints

Info

☑ Arcade Kart (Script)

Script	ArcadeKart

Base Stats

Movement Settings

Top Speed	10
Acceleration	1
Reverse Speed	5

Figure 3.8: This is a sample image of the Unity Hub and our grid road track.

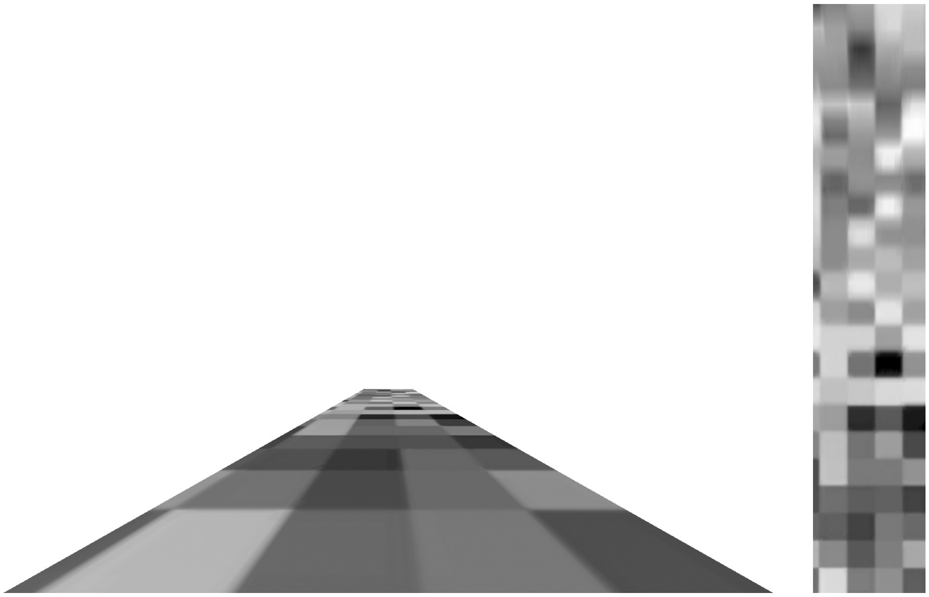
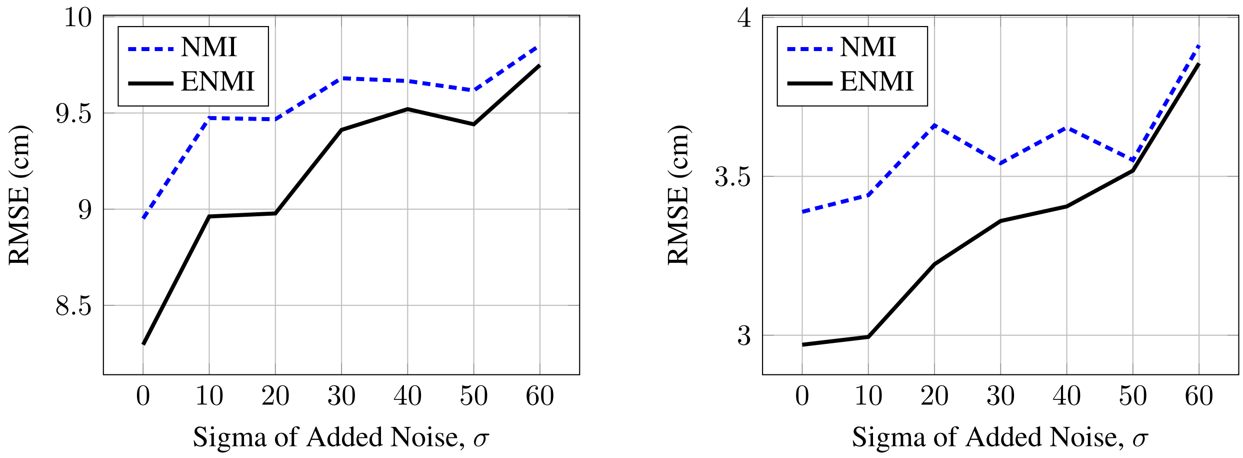


Figure 3.9: The left image shows the cropped portion of the game view from Fig. 3.8. The right image shows the same section transformed into the perspective of the map. Note that this is simply the reverse of Fig. 2.8.

We add Gaussian noise to the game view portion (before transformation) of the images to see how it affects the performance of NMI and ENMI. Figure 3.10 shows the longitudinal and lateral localization RMSE values plotted against the σ of added Gaussian noise. These RMSE plots resemble the right end of Fig. 3.7. This likely means the level of inherent noise in our Unity simulation is comparable to the noise when $N_0 \approx 1$ in Fig. 3.7. The game rendering, screen resolution, and image transformation are all possible contributors to the inherent noise. Both curves in the longitudinal plot have a fairly consistent bias of approximately +5.5 cm across the sigma values; the curves in the lateral plot have a more variable bias ranging from approximately -2 cm to -0.6 cm. Our map resolution is approximately 8 cm per pixel. These biases are likely due to the conversion between pixel position in the map and coordinates in the Unity game.



(a) Longitudinal RMSE

(b) Lateral RMSE

Figure 3.10: This figure shows the longitudinal and lateral localization performance of NMI and ENMI from the Unity simulation.

3.3.2 Ford Image Analysis

We now turn to actual road images to connect our findings with practical AV systems. Both techniques described in this thesis are motivated by the nonuniform noise profile developed in Section 2.3. Though we developed the noise profile based on the physics of our image acquisition model, it is unclear whether it exists in practice. We examine processed LiDAR images supplied by Ford Autonomous Vehicles LLC to discover the noise profile present in an existing AV system.

Figure 3.11 shows a local image capture and its matching prior map image. The local image (left) is generated by combining the point clouds of "four Velodyne HDL-32E 3D-LIDAR scanners" and collapsing them into a top down view of the vehicle and its surroundings [21]. All images have the middle of the vehicle's rear axle at their center. We compute the empirical variance of the pixel values at position $[i, j]$ using

$$\mathrm{Var[i,j]} = \frac{1}{n-1}\sum_{k=1}^{n}\left(a_{ij}^{(k)} - \tilde{a}_{ij}^{(k)}\right)^2,$$

where a and $\tilde{a}$ are matching pairs of local and prior map images [25].

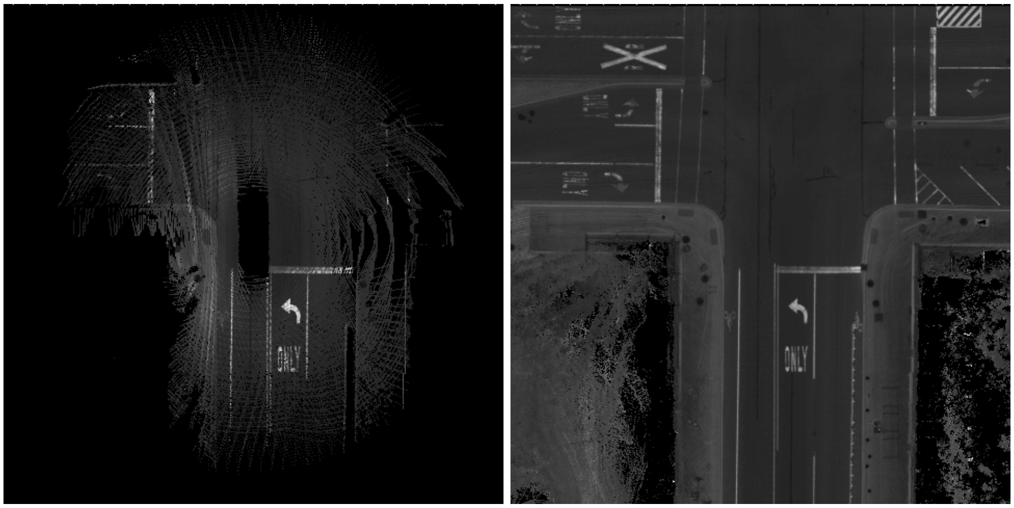


Figure 3.11: The local image (left) is the current sensor input to the vehicle (after processing). The prior map image (right) is the map section that matches with the local image.

We process over 10,000 images to generate the variance mask illustrated in Fig. 3.12. The variance increases radially as you move away from the vehicle; however, it does not behave as described in Section 2.3. The disconnect is expected because the derivations in this thesis are based on an idealized model. Still, the distribution of noise is clearly nonuniform. This validates our assertion that, during localization, one should not give equal confidence to all regions of an image. We can integrate empirical masks, such as Fig. 3.12, into localization algorithms along with ENMI to improve performance.

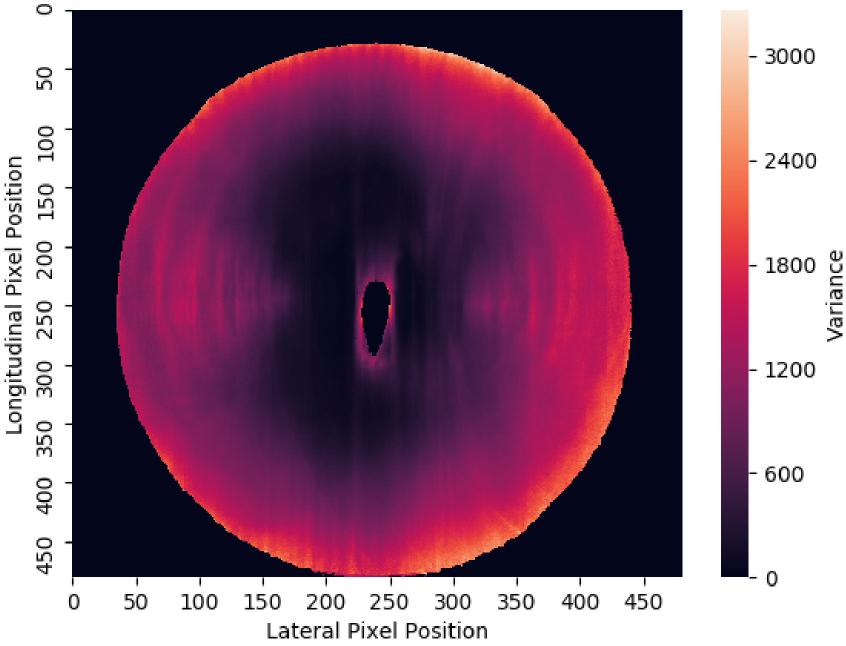


Figure 3.12: This figure illustrates how the empirical variance of the pixel values fluctuates depending on the location relative to the vehicle. Note the front of the vehicle is facing the top of the image. The black region bordering the image is out of range.

3.3.3 Localization Testing with Ford Data

In this section, we evaluate the effectiveness of our proposed ENMI technique using real-world data. Ford Autonomous Vehicles LLC has software to evaluate localization performance offline using LiDAR data captured by their vehicles. Our ENMI technique is integrated into Ford's existing NMI-based localization software by altering the creation of the joint distributions as described in Section 3.2.2. The variance mask shown in Fig. 3.12 is integrated as a 481 by 481 array. We convert the mask to standard deviation values and store it as an array of 8-bit unsigned integers.

We tested our ENMI technique on an approximately 13 minute log of LiDAR data from the Mcity Test Facility in Ann Arbor, Michigan. The data was collected by Ford Fusion testing vehicles equipped "with four Velodyne HDL-32E 3D-LIDAR scanners and an Applanix POS-LV 420 (IMU)" [21]. The ground truth for the vehicle was created by Ford using the raw pose from the Applanix IMU corrected with acquired LiDAR data. The localization software is written in C and run on a Dell Precision 7710. Ford's software, integrated with our ENMI technique, performed offline localization on vehicle pose in 3 degrees of freedom: longitudinal (x), lateral (y), and yaw angle. The longitudinal variable (x) defines the forward and backward position of the vehicle, the lateral variable (y) defines the position of the vehicle left and right, and the yaw angle defines the left-right rotation of the vehicle. The measured errors in these variables are evaluated independently. Figure 3.13 shows the vehicle's calculated trajectory. Table 3.2 shows the longitudinal, lateral, and yaw angle RMSE values resulting from our ENMI-based localization algorithm. The errors over time for each variable are graphed in Fig. 3.14.

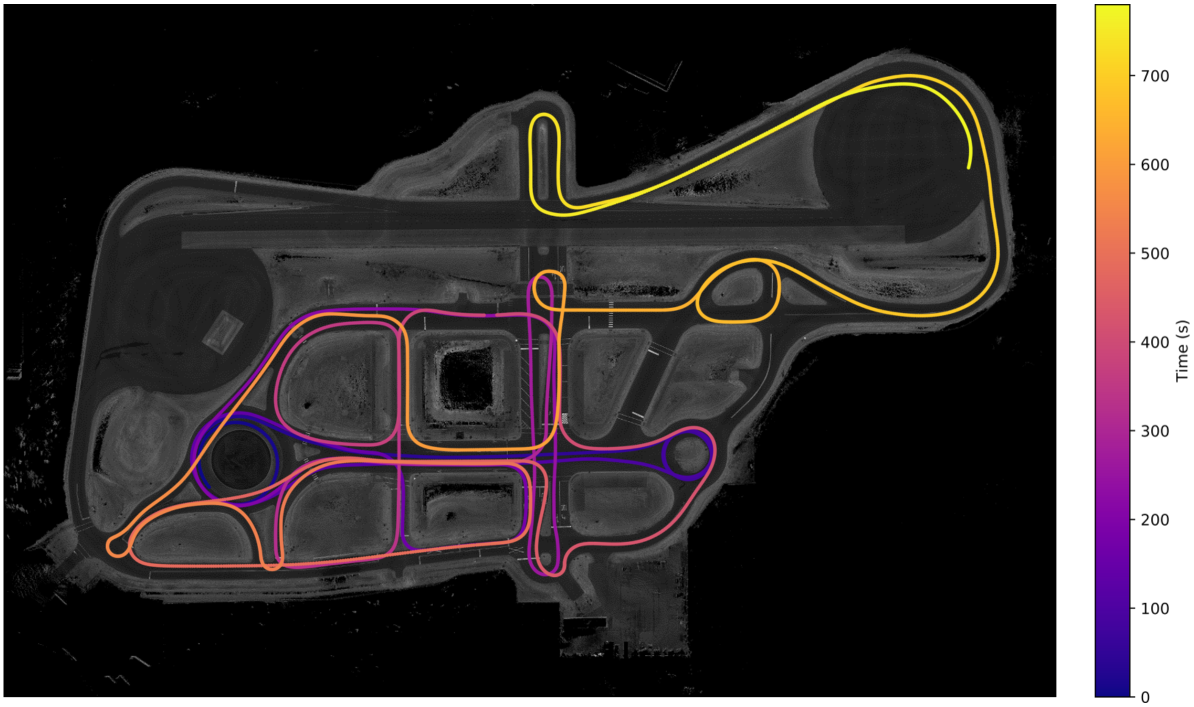
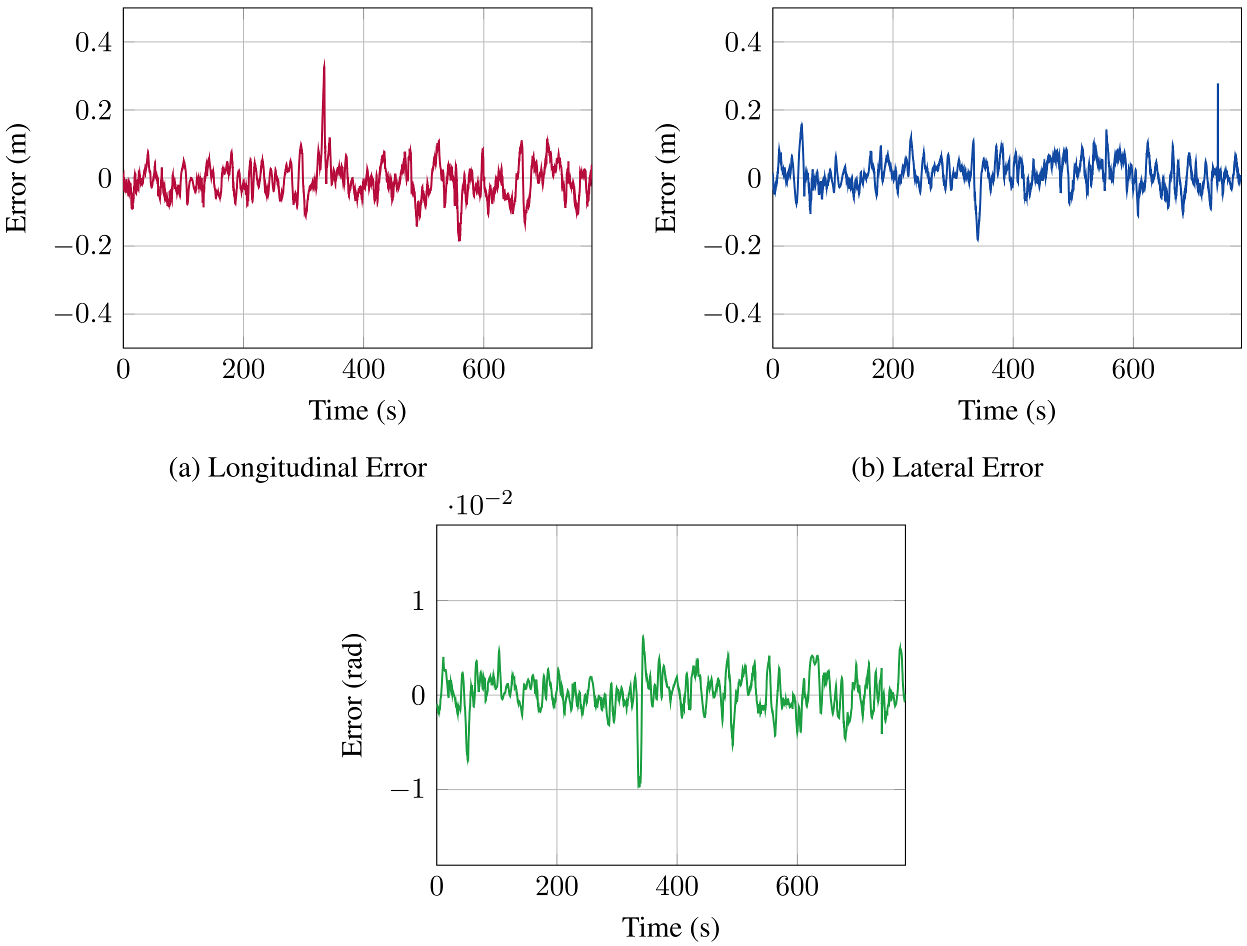


Figure 3.13: This figure shows the vehicle's trajectory as calculated by our localization algorithm.

We do not seek to compare the localization performance of our ENMI technique with Ford's standard NMI-based algorithm because it is currently unclear how well the mathematics of the noise distribution described in previous sections extend to LiDAR data. The purpose of this section is to show that our ENMI technique can be implemented in a state of the art localization platform. It also shows that our technique performs reasonably well on LiDAR data even though it is not completely applicable in this case. The positive results included in this section indicate that our general findings extend to Ford's Fusion AV platform.

Longitudinal (cm)	Lateral (cm)	Yaw Angle (rad)
4.99	4.31	0.00196

Table 3.2: This table states the RMSE values for each variable.



(a) Longitudinal Error

(b) Lateral Error

(c) Yaw Angle Error

Figure 3.14: This figure shows the localization error over time for each variable.

4. SUMMARY AND CONCLUSIONS

The localization task is integral to a successful autonomous vehicle. In this thesis, we described a noise profile unique to autonomous vehicles that is not currently accounted for in localization strategies. We then proposed two techniques for localization that account for such noise: the weighted inner product and enhanced NMI. The performance of each technique was evaluated through numerical simulations. ENMI was further evaluated through Unity simulations and testing with real-world data provided by Ford Autonomous Vehicles LLC.

Our localization techniques performed well in numerical simulations, outperforming their counterparts especially in the presence of noise. The results of the Unity simulation were also positive with ENMI outperforming NMI in both the longitudinal and lateral directions. Additionally, ENMI integrated successfully into Ford's localization software and performed well, though slightly worse than the state of the art. This is likely due to mismatch between our pinhole camera model and the array of LiDAR sensors on Ford's vehicles.

4.1 Challenges

The main challenge in this work was integrating the ENMI technique into Ford's localization software. Due to its proprietary nature we only had access to the portion of the code we were editing. Additionally, we did not have access to the localization testing environment in which the software runs. Changes to the code were made locally then tested at Ford Autonomous Vehicles LLC. We worked closely with Siddharth Agarwal in this effort and are very grateful for his time and contribution to this work.

Other challenges included building the Unity simulation environment and running numerical simulations in a timely manner. The Unity development platform is not regularly used for localization testing. We had some difficulty building the grid road track and getting a consistent response from the driving controls. Some of the numerical simulations included in this thesis were too computationally intensive to run on a personal computer. These simulations were run with the help of

Texas A&M High Performance Research Computing.

4.2 Future Work

There are many potential projects either branching off from or directly following our work. Optimizing camera position and orientation for localization performance was suggested by a reviewer of our inner product paper [24]. We have discussed some interesting ways to makes roads more feature-rich and thus better suited for precise localization. Both of these projects could produce interesting and impactful results.

Following directly from our work it would be beneficial to develop a better testing platform, either physical or simulated, to more precisely evaluate ENMI. Most state of the art AV platforms rely on LiDAR sensors which make them an imperfect application for ENMI. A clear next step for our research that circumvents this problem is to derive a noise characterization for LiDAR point clouds and adjust ENMI accordingly. This would allow us to compare our localization performance with the state of the art in an applicable setting.

We would also like to explore solutions to the dirty lens problem. This is when the lens of a LiDAR sensor or camera becomes dirty during operation and partially blocks the vehicle's view of its surroundings. It would be very interesting to apply our variance mask from Section 3.3.2 to this problem. One possible solution is a dynamic variance mask that updates its values to account for sections of the current sensor input blocked by debris on the lens.

REFERENCES

[1] N. H. T. S. Administration, “National statistics,” 2020.

[2] N. S. Council, “Road safety,” 2020.

[3] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, “A solution to the simultaneous localization and map building (slam) problem,” *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, 2001.

[4] A. Eliazar and R. Parr, “Dp-slam: Fast, robust simultaneous localization and mapping without predetermined landmarks,” in *IJCAI*, vol. 3, pp. 1135–1142, Acapulco, Mexico, 2003.

[5] M. Bosse, P. Newman, J. Leonard, and S. Teller, “Simultaneous localization and map building in large-scale cyclic environments using the atlas framework,” *The International Journal of Robotics Research*, vol. 23, no. 12, pp. 1113–1139, 2004.

[6] S. Thrun and M. Montemerlo, “The graph slam algorithm with applications to large-scale mapping of urban structures,” *The International Journal of Robotics Research*, vol. 25, no. 5-6, pp. 403–429, 2006.

[7] J. Levinson, M. Montemerlo, and S. Thrun, “Map-based precision vehicle localization in urban environments,” in *Robotics: Science and Systems*, vol. 4, p. 1, Citeseer, 2007.

[8] J. Levinson and S. Thrun, “Robust vehicle localization in urban environments using probabilistic maps,” in *2010 IEEE International Conference on Robotics and Automation*, pp. 4372–4378, IEEE, 2010.

[9] J. Fuentes-Pacheco, J. Ruiz-Ascencio, and J. M. Rendón-Mancha, “Visual simultaneous localization and mapping: a survey,” *Artificial Intelligence Review*, vol. 43, no. 1, pp. 55–81, 2015.

[10] L. Jetto, S. Longhi, and G. Venturini, “Development and experimental validation of an adaptive extended kalman filter for the localization of mobile robots,” *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 219–229, 1999.

[11] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever, “Mutual-information-based registration of medical images: A survey,” *Trans. Medical Imaging*, vol. 22, no. 8, pp. 986–1004, 2003.

[12] J. V. Hajnal and D. L. G. Hill, *Medical Image Registration.* Boca Raton: CRC press, 2001.

[13] A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens, and G. Marchal, “Automated multi-modality image registration based on information theory,” in *Information Processing in Medical Imaging*, vol. 3, pp. 263–274, 1995.

[14] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, “Multimodality image registration by maximization of mutual information,” *IEEE transactions on Medical Imaging*, vol. 16, no. 2, pp. 187–198, 1997.

[15] W. M. Wells III, P. Viola, H. Atsumi, S. Nakajima, and R. Kikinis, “Multi-modal volume registration by maximization of mutual information,” *Medical Image Analysis*, vol. 1, no. 1, pp. 35–51, 1996.

[16] P. Viola and W. M. Wells III, “Alignment by maximization of mutual information,” *International Journal of Computer Vision*, vol. 24, no. 2, pp. 137–154, 1997.

[17] W. R. Crum, T. Hartkens, and D. Hill, “Non-rigid image registration: theory and practice,” *The British journal of radiology*, vol. 77, no. suppl_2, pp. S140–S153, 2004.

[18] C. Studholme, D. L. G. Hill, and D. J. Hawkes, “An overlap invariant entropy measure of 3d medical image alignment,” *Pattern Recognition*, vol. 32, no. 1, pp. 71–86, 1999.

[19] A. Dame and E. Marchand, “Mutual information-based visual servoing,” *Trans. Robotics*, vol. 27, no. 5, pp. 958–969, 2011.

[20] R. W. Wolcott and R. M. Eustice, “Visual localization within Lidar maps for automated urban driving,” in *International Conference on Intelligent Robots and Systems*, pp. 176–183, IEEE, 2014.

[21] J. Castorena and S. Agarwal, “Ground-edge-based lidar localization without a reflectivity calibration for autonomous driving,” *Robotics and Automation Letters*, vol. 3, no. 1, pp. 344–351, 2017.

[22] M. Young, “The pinhole camera: Imaging without lenses or mirrors,” *The Physics Teacher*, vol. 27, no. 9, pp. 648–655, 1989.

[23] D. Hearn, M. P. Baker, and W. R. Carithers, *Computer Graphics with OpenGL*, ch. Perspective Projections, pp. 327–340. Prentice Hall, 2011.

[24] S. T. Flanagan, D. K. Khublani, J.-F. Chamberland, S. Agarwal, and A. Vora, “Localization in autonomous vehicles using a generalized inner product,” in *2019 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 1–5, IEEE, 2019.

[25] S. T. Flanagan, D. K. Khublani, J.-F. Chamberland, S. Agarwal, and A. Vora, “Enhanced normalized mutual information for localization in noisy environments,” in *2020 IEEE Applied Signal Processing Conference (ASPCON)*, pp. 178–182, IEEE, 2020.